inContexto: A Fusion Architecture to Obtain Mobile Context

Gonzalo Blázquez Gil, Antonio Berlanga de Jesús and José Manuel Molina López
Group of Applied Artificial Intelligence
University Carlos III of Madrid
Colmenarejo, Spain
Email: gbgil@inf.uc3m.es, {molina,aberlan}@ia.uc3m.es

***Abstract*—Thanks to the embedded sensors providing in mobile devices will revolutionize the way to carry out with. Mobile devices provide a set of embedded sensors, such as accelerometer, digital compass, gyroscope, GPS, microphone, and camera. Another point to consider, is that mobile devices are easily programmable since an API was included by the OS companies. This paper aims to describe a distributed architecture, called inContexto, to recognize physical actions perfomed by users such as walking, running, being stand, sitting and also retrieve context information from the user. Sensory data is collected by HTC magic application made in Android OS.**

Keywords: Mobile device, Context-aware, Information fusion, Hard Sensors, Soft sensors.

I. Introduction

In 2009, smartphone penetration in the US was 25% and 14% of worldwide mobile phone shipments were smartphones [23]. By 2011, its sales are projected to overcome desktop PCs. Hence, smartphone is becoming increasingly popular as a personal computer and they are becoming the main computer and communication device in people's lives.

Obtaining context aware from smartphones presents several problems [9] [12]. The principal problem is that they are not built to collect information and infer activities. Nevertheless, smarthphones experience almost the same physical forces, temperature and noise that the person who carries them out. So, if you track mobile phone actions you are tracking person actions. In [26], a mobile phone is adduced as a context aware device and an activity recognition sensor.

Context-aware systems allow th development a new kind of mobile applications which may be represented as a context based scenario where there are individuals who require a satisfaction of their needs and there are providers which could solve these lacks. In context-aware computing, context is any information that can be used to characterize the situation of an entity.

The inference of user context implies a large number of sensors distributed over the body and/or environment, depending on the activities to detect [14]. Smartphones are especially well-suited to this task because they are integrated with Microelectromechanical systems (MEMS) which make easier to obtain user information. At the same time a wearable system must be inconspicuous and operate during long periods of time. This implies minimizing sensor size, and especially low energy consumption [15]. For that reason, it may be possible to consider a Smartphone like a non-intrusive device to obtain context aware from people.

Mobile devices may obtain and process physical phenomena from embedded sensors and send this information to remote locations without any human intervention [17]. Mobile devices should take advantage of mobile contextual information, such as position, user profile or device features; to offer greater services.

In addition to microphones and digital cameras, nowadays, every mobile device has several sensors embedded: accelerometer, gyroscope, compass, magnetometer, proximity sensor, light sensor, GPS, etc; which provide to developers thousand of new data to improve user mobile experience [9]. For example, the iPhone have a calculator app which uses orientation sensor to restor to user interface. The app in portrait mode has only the basic functionality however in landscape mode the app turns up a scientific calculator. In the future more and more sensors are being incorporated into smartphones and nobody knows which will be the limit, humidity sensor, barometer, etc.

Sensor networks have been contributed to numerous attractive applications in areas such as military, environmental monitoring, human activity recognition, eHealh [18] [24] [4].

These applications usually employ hard sensor (accelerometer, gyroscope, etc.) to infer personal context, but there are another kind of sensor called Soft sensors which are well-fit to obtain user information. Frequently, soft sensors are referred as human observer that provides his/her point of view of something. Given information by the human is not precise, for example, someone is going somewhere or doing something. At this point Social Networks Sites (SNS) play an important role. Users daily share their personal information on Facebook, linkid, twitter and so on. Using the information posted on SNS is possible to infer personal data from users.

Information fusion systems provides many advantages to retrieve context aware information. However, to handle all the information from the different sensors, either if it is hard or soft sensors, is pretty costly. In an extreme case, each sensor may have its own processor to fuse the local data and cooperate with other sensor nodes.

This paper is focused on the descrition of inContexto, an Information Fusion Architecture to identify context aware from mobile devices as well as user who carries it. This architecture is rely on the Information Fusion advantages, using social networks as a soft sensors and MEMS on the mobile device as hard sensors.

II. Related works

A mobile sensing architecture to obtain user physical activities and to share them on social network is presented in [13] , this architecture is called Cenceme. However, Cenceme does not use other information (Context aware information) to complete the taken actions. The proposed architecture is split in three layers: Sense, learn and share. Sense layer aims to collect raw sensor data from sensors embedded in the phone.

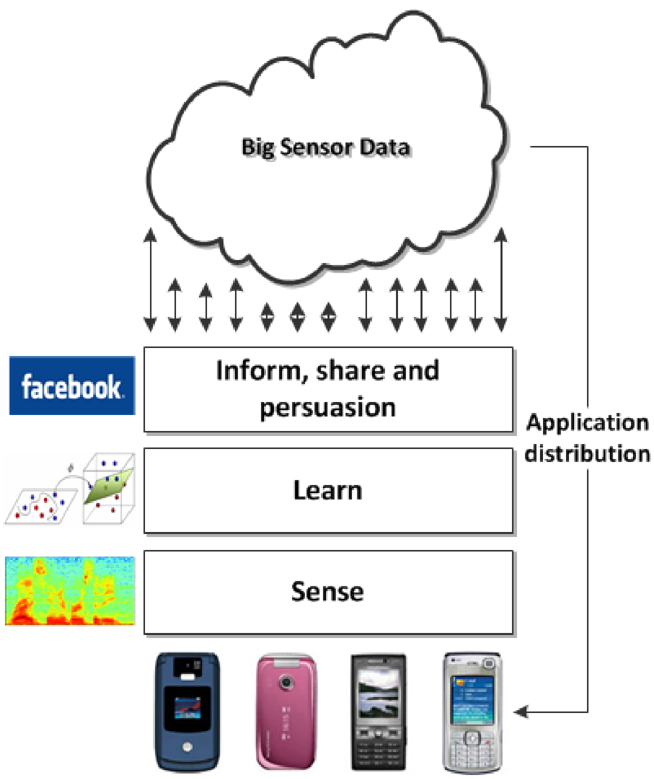

Figure 1. Proposed Architecture by Cenceme(Sense, Learn and Share layer).

In learn layer, they propose to use a variety of data mining techniques to infer user rules. These techniques are used to interpret mobile data extracted from sensor layer. Their approach is to share information in a web portal where sensor data and inferences are easily displayed.

Yohan Chon et all. [3] present LifeMap, an Smartphone-based Context Provider for Location-based Services. The presented architecture is splitted in four component: (i) All the sensors are placed on the low level, this level sends the obtained information (ii) to the Component Manager where information is processed and provide high-level information. Using high-level information from the Component Manager, (iii) the Context Generator generates a point of interest (POI) which contains the user context. The context map is stored in a database to match and aggregate user contexts. And finally, (iv) The Database Adapter is an interface to provide user context to other applications.

Multisensor Fusion architectures are not common in Smartphone applications. However,an architecture for lifestyle monitoring is presented in [9]. The presented architecture just collects data from sensors in the Smartphone and subsequently, information is sent to a PC for data analysis.

Information Fusion is necessary to integrate the data from the different sensors and extract the relevant information on the users. Normally, data fusion architectures are based on an centralized system. In this case, due to low energy consumption problem, it is necessary to design a distributed architecture and to share computational process between Smartphone and cloud servers.

Historically, data fusion methods were developed basically for military applications. The military community has developed a layout of functional architectures based on the Joint Directors of Laboratories model for multisensor systems. However, in recent years these methods have been applied to civilian applications [10].

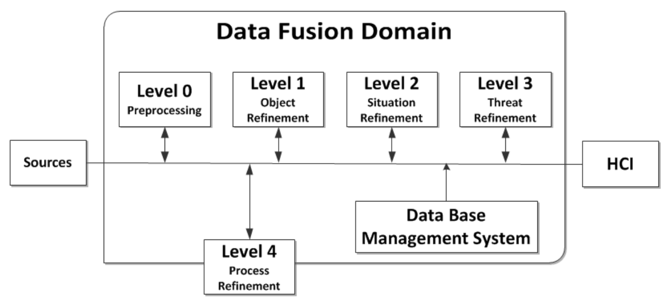

Figure 2. JDL Information Fusion model.

The JDL model was never intended to decide a concrete order on the data fusion levels. Levels are not alluded to be processed consecutively and it can also be executed concurrently. Figure 2 depicts JDL data fusion process high level model.

- Level 0: sub-object data assessment, is associated with pre-detection activities such as pixel or signal processing, spatial or temporal registration.
- Level 1: At this level, identify and locate objects is attempted. Hence, it is reported the object situation by fusing the attributes from diverse sources. The steps included at this stage are:
  - Data aligment: the data is processed to acquire a common spatial and time frame.
  - Data association: In this step is measured the degree of proximity among variables or the same variable.
  - Object estimation: At this stage the data fusion centre estimates the object's position, velocity, or attributes.
  - Object identity: And finally, it is made a prediction of the object's identity or a new classifictaion object is made.

- Level 2: Attempts to construct a picture from incomplete information provided by level 1, that is, to relate the reconstructed entity with an observed event.
- Level 3: Interprets the results from level 2 in terms of the possible opportunities for operation. It is analysed pros and cons of taking one action over another one.
- Level 4: Process refinement is an element of resource management and used to close the loop by re-tasking resources (e.g. sensors, communications and processing) in order to support the objectives.

The waterfall IF model was proposed by Markin et all [16] (see Figure 3. This architecture emphasizes on the processing functions on the lower levels. However, waterfall model ommits any feedback data flow instead of JDL model which every level is interconnected. The relationship between waterfall architecture and JDL model is as follows:

- Sensing and signal processing correspond to level 0.
- Feature extraction and pattern processing match with level 1.
- Situation assessment is similar to situation refinement in JDL model (level 2).
- And finally, decision making corresponds to the third JDL level 3.

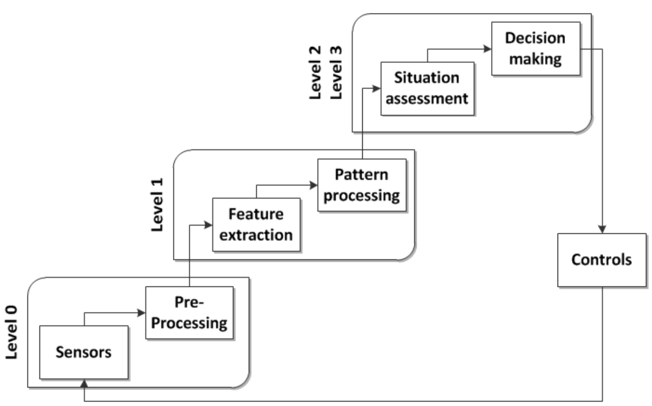

Figure 3. Waterfall Information Fusion model.

III. Describing Mobile Device Context

First of all, in order to use context correctly, it is crucial to define what researchers think context is. The term pervasive computing was introduced by Waiser in 1991 as: *the seamless integration of devices into the users everyday life*. In general context aware is represented by applications which change their behavior according to the conditions around them, in this case the mobile device conditions. It is convenient that applications and services react specifically to their surrounding, location and time. Summarizing, their behavior is able to change according to circumstances.

In 1994 was introduced the term context aware computing by Schilit and Theimer [20]. They defined context as software that adapts according to itself location of use, the collection of nearby people and objects, as well as changes to those objects over time. Subsequently, some other researchers try to formally define context, for example, Schmidt et al. [21] define context as knowledge about the user's and IT device's state, including surroundings, situation and location. One of the most accurate definition was given by Dey [6]. He defined context as:

Any information that can be used to characterize the situation of entities (i.e., whether a person, place or object) that are considered relevant to the interaction between a user and an application, including the user and the application themselves.

Besides, he defined the three kind of entities: places (buildings, rooms, etc), people (individual or groups of people) and finally things (electronic devices, physical objects, etc). Each entity is characterized by four categories: Identity which identify each entity, location which provides entity's position, status or activity and the last one is time.

According to Dey's definition about context and entity, this work presents how to obtain context information from mobile devices.

1) Identity: In order to identify one person is possible to use different sources, hardware or software. Hardware identification, as MAC address, presents several problems because you are identifying the mobile device and you are not identifying the person. Hence, if another user manipulates the same device, there will be identification problems. However, using software identification as a facebook or google account (OpenID) this problems will be solved. OpenID is an open standard that describes how users can be authenticated in a decentralized manner, obviating the need for services to provide their own ad hoc systems and allowing users to consolidate their digital identities [19].
2) Location: Location aware could be the main factor in the development of, context applications. Nevertheless, Location-aware is only one aspect of context aware as a whole [22]. Location context may be described as an application dependent on the geographical location. Location answer the question where is the action taking place?. For example, it is possible to define a running action, however, it could be interesting to define where is he/she running? and where is he/she running to?. In outdoor environments GPS provides a good solution to determine the location of mobile devices however in GPS-denied areas such as urban, indoor, and subterranean environments, unfortunately, an effective solution does not exist. Besides, every location system provides in its own way location data. Recently, W3C has releaded a Geolocation API [8] to standardize an interface to get back the geographical location information for a client device.
3) Status or Activity: Talking about status it is necessary to differentiate user status and mobile phone status. Mobile device status mainly refers to communication behavior: Calls and calls attempts, sent and received SMS, SMS content, battery level, wireless connections, etc. On the contrary, user status does not refer just to her/his

calendar (working, sleeping, free-time, etc.) otherwise the relevant information about the user, normally, it is included in the user profile as an instance (name, date of birth, where is she/he born, etc). As it was described previoulsy people movement are reflected in mobile devices sensors. The generated information can be used to identify different activities (e.g., running, walking, standing, cycling and so on) that the user is performing. These kind of actions are obtained by low level sensors provided by the mobile phone (Accelerometer, Gyroscope, light sensor, microphone, etc.). For example, accelerometer is able to describe the physical movements of the user carrying the phone.

4) Time: Activities taken by the user or the user's status do not have any meaning if it is impossible to set the action in a place and in time. For that reason time is an essential in context aware applications.

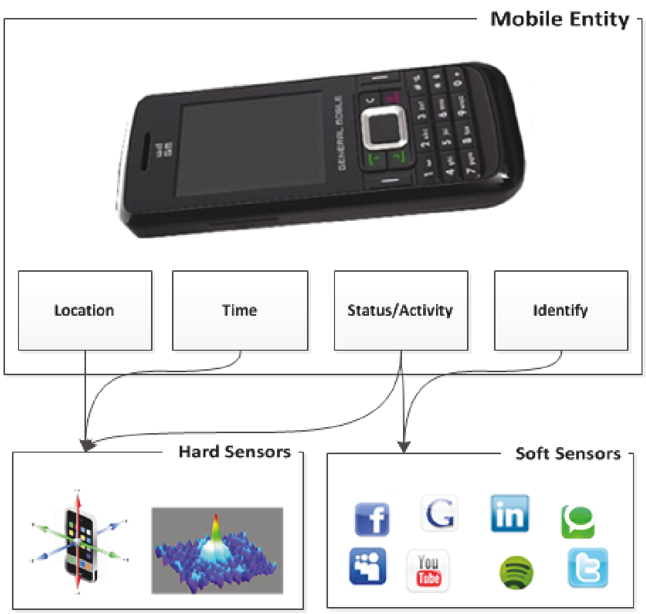

Figure 4. Entity concept from Mobile Context sources.

IV. Describing sources of mobile context

The proposed architecture was tested in an HTC Magic with Android operating system. Andorid OS provides four sampling frequencies (50Hz, 20Hz, 10Hz and 4Hz) to collect data from MEMS, micro-electromechanical sensors. The low sampling rate is due to the Android OS restrictions and there is no control over it. In this paper, it is used the highest sampling frequency provided but it is necessary to research the impact in the battery life.

A. Hard sensor

Advances in miniaturization permit sensors to be embedded in mobile devices. The camera and microphone are probably the most used sensors in Context aware system. However, these sensors present several issues, such as to retrieve user information. Mobile devices are mostly in the pocket and would affect the quality of the captured sensor data.

Basically, using this kind of sensors it is possible to obtain basic actions taken by the user such as: running, walking, standing, talking, listening music, etc. These actions are obtained by low level sensors provided by the mobile phone (Accelerometer, Gyroscope, light sensor, microphone, etc.). For example, accelerometer is able to describe the physical movements of the user carrying the phone.

Sensors are already integrated in many modern mobile devices such as Apple iPhone, iPad, Android devices, etc. Normally, embedded sensors are placed in the same chipset. In this case, it is used a HTC Magic mobile device is AK8976A marketed by Asahikasei Microsystems Co., Ltd(AKM). This chipset includes a 6-axis electronic compass that combines a 3-axis geomagnetic sensor with a 3-axis acceleration sensor in an ultra-small package. Consequently, whether you applications query the accelerometer, compass or both, it consumes the same energy power.

1) Accelerometer: A tri-axial accelerometer is a sensor that returns a real valued estimate of acceleration along the x, y and z axis from which velocity and displacement can also be estimated. Accelerometers can be used as motion detectors as well as for body-position and posture sensing [5]. Collected data from the accelerometer has the following attributes: time, acceleration along three axis (x, y and z), not including gravity.

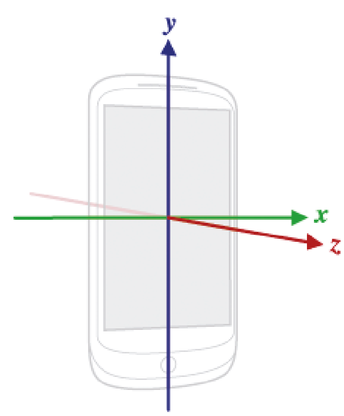

Figure 5. Mobile device coordinates origin.

Accelerometer provides data from the origin of coordinates of the device which is placed in the lower-left corner with respect to the screen, with the X axis horizontal and pointing right, the Y axis vertical and pointing up and the Z axis pointing outside the front face of the screen. In this system, coordinates behind the screen have negative Z values. Hence if the mobile device is worn on a pocket, it is not clear which axis or axes represent the vertical movement. In the next section, how to transform, using digital compass, coordinates from mobile device representation to real world one will be described.

2) Digital Compass: The Digital compass provides two measures, the first one is the orientation. Values are in radians/second and measure the rate of rotation around the X(roll), Y (pitch) and Z (yaw or Azimuth). The coordinate system is the same as is used for the acceleration sensor. The second one is the magnetic field which measure the ambient magnetic field in the X, Y and Z axis.

Digiatl Compass reports the angle between the magnetic north and the mobile phone's Y axis (orientation measurement). All values are in micro-Tesla (uT) and it measures the ambient magnetic field in the X, Y and Z axis.

3) Gyroscope: Gyroscopes are the most commonly used sensors for measuring angular velocity and angular rotation in many navigation and homing applications. They measure how quickly an object rotates, specifically, measure the rate of rotation around the X, Y and Z axis. The coordinate system is exactly the same as is used for the acceleration sensor. Gyroscopes are the only inertial sensors that provide measurement of rotations without being affected by external forces, including magnetic or gravitational or fabrication imperfections.

4) Location sensor: There are two ways to locate the mobile device, first of all using a GPS, in this case every Smartphone provides an Assisted GPS [25]. A-GPS improves the performance by adding information, through another data connection (Internet or other), than unassisted GPS. In order to receive and process signals is computational costly, minimizing the amount of time and information required from the satellites. The A-GPS receiver uses satellite to locate itself but it can do more quickly and using weaker signals than an unassisted GPS. Normally, an A-GPS provides 2-4 meters error.

The second way to locate the mobile device is using GSM cell tower triangulation. This technique reduced as well ss accurate than GPS however, the energy consumption is reduced as well. According to the application goals, it is necessary to balance the accuracy and the energy consumption and it could be enough a coarse location (GSM) instead of a precision location (GPS).

B. Soft sensor

Social Networks Sites Sites (SNS) are increasingly popular these days. Since their introduction SNS have seduced millions of users. MySpace, Facebook, linkedId and Twitter are the most popular sites. Every site seduces diverse audiences, while Linkedid is focus on work relationships, facebook preferences are friendships. In [2] is described Social Network Site as:

web-based services that allow individuals to (1) construct a public or semi-public profile within a bounded system, (2) articulate a list of other users with whom they share a connection, and (3) view and traverse their list of connections and those made by others within the system. The nature and nomenclature of these connections may vary from site to site.

Each SNS is implemented with specific features, however all of them have a common point which consist of visible profiles. Daily, SNS users share their personal information, SNS manage an uncountable gigabytes of useless user information. Why do not we use these data to obtain user information?

Tipically, User profiles include descriptors such as age, location, interest schools attended. User profiles are becoming more precise: Music preferences, movies, clothes, friendship relationships, personal agenda, etc.

Therefore, this paper proposes to generate its own user profile linking each Social Network Site supported. At first, the chosen SNS are Facebook and LinkId.

V. InContexto: Architecture Definition

In this section is described inContexto, an architecture to obtain context from a user who carries out an smartphone. Mobile phone sensing is still in its infancy, it is not clear what architectural components should run on the device and what should run on the cloud.

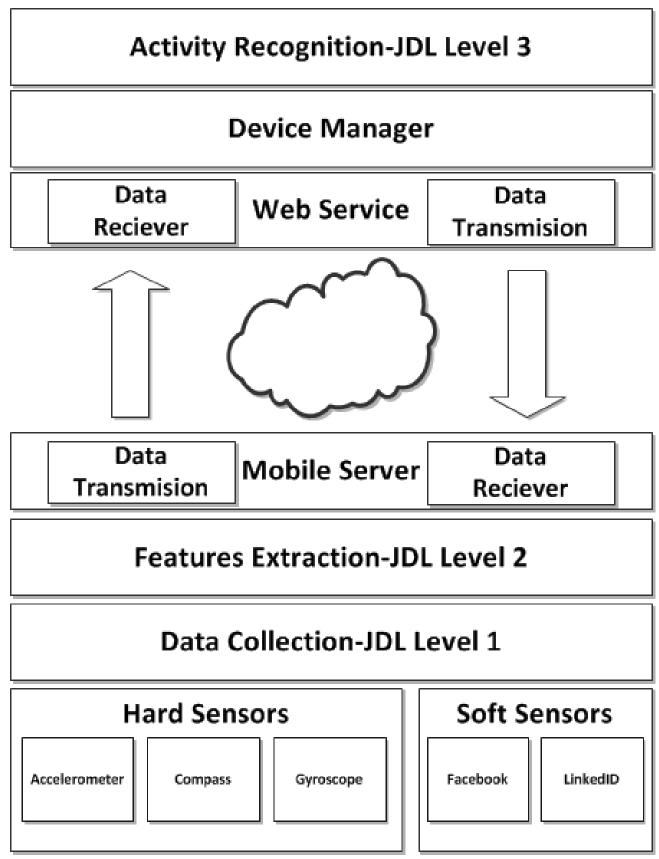

Figure 6. Matching between context and actions.

Sensing, Data Collection and Features extraction are implemented in the mobile phone, on the contrary, the activity recognition is executed in backend infrastructure. Acording to JDL model, inContexto, is a layered architecture where the first three levels running in the mobile phone (an entity) represents the first three levels of JDL model (0, 1 and 2 level) and the last one (Activity recognition level) simbolizes the third JDL model level. In a JDL information fusion model all the components are interconnected.

inContexto is implemented following distributed architecture, a comunication component is designed to associate mobile phones and backend server. Besides, this architecture (Figure 6) needs to accommodate both contextual information as well as hard and soft sensors.

A. Sensors-Level 0

This level aims to collect every single raw data from sensors :Accelerometer, Gyroscope, GPS and soft sensors. It is largely to recall that the presented architecture is developed to obtain context in an non-intrusive way. Although, previous work suggest that the best place to wear this sensor is the hip [1], however, mobile devices used to be stored in different places. For that reason, the data collection process was made while the mobile device was worn in a trousers pocket.

1) Hard sensors: Hard sensor data is accessed through Android API, in concrete SensorManager class which provides methods to obtain all the mobile sensors. A low-level sensing module continuously gathers relevant information about the user activities using sensors thanks to Android provides background non-user interactive processing using a Service model. In this approximation the architecture just acquire data from Digital Compass, Accelerometer and Gyroscope.

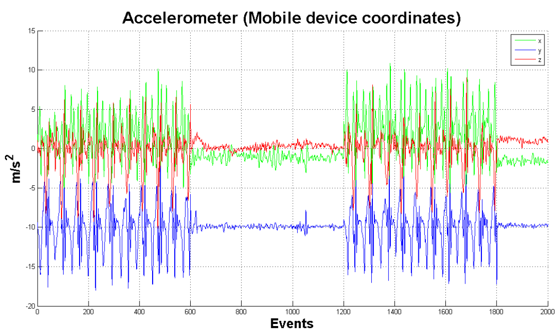

Figure 7. Device 3-axes accelerations.

The sampling frequency can be adjusted according to the action studied. In this case, rely on the next study [11], the sampling frequency range requiring to obtain human actions is $0.6Hz$ to $2.5Hz$. Consequently, to prevent aliasing problem, the Nyquist-Shannon sampling theorem is followed:

$$F_E \geq 2 * F_{Max}$$

Finally, the sampling frequency was fixed to 50 Hz, to maintain sufficient accuracy.

2) Soft sensors: Adquiring context from soft sensors is not a banal work. Social Network information is accessed thanks to an API which provides an interface. Latley, user login is necessary.

Social networks are plenty of information and most of this information is unnecessary. Thus, the selected features collected from different social networks are: Social Network id, Social network name, born on, lives in and a friendships array which contains the same information excluding the friendship array.

B. Data Collection-Level 1

This component represents the first level in JDL architecture. Thanks to every single data is collected in the same device, Data Collection level does not present a problem.

1) Data Collection from hard sensors: In order to manage the important information from the sensor, the collected raw data is filtered. Accelerometer data is related mobile device coordinates. However, it is necessary to tranform that forces according to the real world. Computes the inclination matrix I as well as the rotation matrix R transforming a vector from the device coordinate system to the world's coordinate system which is defined as a direct orthonormal basis. I matrix is a simple rotation around the X axis and the rotation matrix R which is the identity matrix when the device is aligned with the world's coordinate system.

$$a_{realworld} = a_{mobiledevice} * R * I$$

Figure 7 represents the device accelerations and shows the changes of the three forces depending on the movement taken it by the user (Running, Walking, Standing). On the other hand, Figure 8 represents the transformation from the mobile device reference to the real world reference.

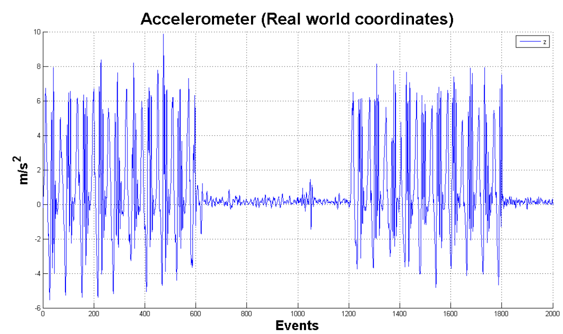

Figure 8. Real world vertical acceleration.

2) Data Collection from soft sensors: In this level, information from different social networks is fused to manage only one structure which contains the user information. Basically, the selected features from the level 0 are related to personal information and friendships.

This information is proccesed and it creates a meta-structure which contains level 0 information. The selected structure to manage these information is an Ontology. It since it permits to add new features in consecutive iterations.

C. Features Extraction-Level 2

The features extraction level is also implemented in the mobile phone. This level aims to process and select which features are better to identified an action. The module processes the raw sensor data into features that help discriminate between activities. Considering computational power and energy consumption restrictions of mobile phones It is used the spectrogram (Figure 9) as a features extraction technique.

A spectrogram is a time-varying spectral representation that shows how the spectral density of a signal varies with time. Spectrograms was calculated from real world acceleration time data (Figure 7) using the Short-Time Fourier transform (STFT).

$$STFT(t^1, f) = \int_t x(t) * e^{-2\pi i f t} * w(t - t^1) dt$$

where $x(t)$ is the signal to analyze and w apodization function (Hanning).

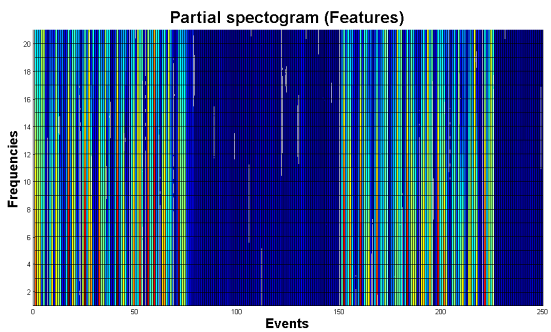

Figure 9. Complete and partial spectrogram (Features selection).

The active frequencies (Red and yellow bars) depending on the action taken (Steady actions or not steady action) are clearly distinguishable. Non active frequencies are colored in blue which represent when the user is doing a sedentary action (Standing, sitting, etc). On the contrary if the spectrogrtam is colored, the user is taking an active action (Running, walking, etc.).

D. Mobile Server, Web Service and Device Manager

Both components aim to communicate the mobile device with the server. One of them (Mobile server) is implemented in the mobile devices and the other one (Web service) is on the server. The Web service module is developed as web service which is designed to support interoperable machine-to-machine communication over a network. Web-services provide an interface which describe message format, specifically, Web-Services Description Language WSDL [7]. Device manager allows web-services to view and control the devices attached to the service. When a device is not online, the web-server keep the last device's IP address for a while, waiting for a new connection.

E. Activity recognition/Level 3

The last layer is classification module that uses the features selected in the mobile phone to infer what activity an individual or group of individuals are engaged in. In this component, it will be implemented the algorithms (Supervised learning, Probabilistic classification, Model-based or instance-based learning) to figure out the final taken action. The training proccess will be run off-line on desktop machines since it is computationaly costly.

The activity recognition level fetches the features selected by the second level and classifies this features in order to return the current activity: Walking, running, sitting, standing.

For example, The accelerometer and the microphone may detect whether the user is sitting or the user is near a sound source. If you use the both actions and it is able to locate the action (living room) it could figure out that the person is sitting in the living room watching TV (Location action).

VI. Conclusions

In this paper, it was presented inContexto, a distributed architecture to obtain mobile context from mobile devices. The proposed architecture distributes the processing load between mobile device and a server placed on the cloud. With this approach the energy consumption is reduced, increasing autonomy to offer a better service to the user.

Besides, a mobile device entity is defined according to Dey's definition of context aware. An entity is defined as a mobile device which provides hard or/and soft sensors, provides internet connection everywhere and is portable.

Activity recognition systems identify and record in real-time selected features related on user activity using a mobile device. The paper describes how to face with this problem using an information fusion architecture in mobile devices. Besides, it describes sensing module process, that is one of the most important components in activity recognition systems.

Considered future works extending the development of the server module and also it will extend activity classifier to more complex activities (Group activities, Interaction activities). Context information will be used to infer the user's emotional state, for example, according to the social network state, the music which is listening at the moment, the place where the user is and using another hard and low sensor.

Another important line of research is to measure the accuracy of the activity recognition techniques depending on the sensor sampling frequency. The highest frequency was used in this preliminary study.

VII. Acknowledgments

This work was supported in part by Projects CICYT TIN2008-06742-C02-02/TSI, CICYT TEC2008-06732-C02-02/TEC,CAM CONTEXTS (S2009/ TIC-1485) and DPS2008-07029-C02-02.

References

[1] L. Bao and S. Intille. Activity recognition from user-annotated acceleration data. *Pervasive Computing*, pages 1–17, 2004.

[2] D. Boyd and N. Ellison. Social network sites: Definition, history, and scholarship. *Journal of Computer-Mediated Communication*, 13(1):210–230, 2008.

[3] J. Chon and H. Cha. LifeMap: Smartphone-based Context Provider for Location-based Services. *IEEE Pervasive Computing*, 2011.

[4] S. Consolvo, D. McDonald, T. Toscos, M. Chen, J. Froehlich, B. Harrison, P. Klasnja, A. LaMarca, L. LeGrand, R. Libby, et al. Activity sensing in the wild: a field trial of ubifit garden. In *Proceeding of the twenty-sixth annual SIGCHI conference on Human factors in computing systems*, pages 1797–1806. ACM, 2008.

[5] R. DeVaul and S. Dunn. Real-time motion classification for wearable computing applications, 2001.

[6] A. Dey and G. Abowd. Towards a better understanding of context and context-awareness. In *CHI 2000 workshop on the what, who, where, when, and how of context-awareness*, volume 4, pages 1–6. Citeseer, 2000.
[7] W. W. W. C. Editors. Web services description language 1.1 @ONLINE, Mar 2001.
[8] W. W. W. C. Editors. Geolocation api specification @ONLINE, Feb 2010.
[9] R. Ganti, S. Srinivasan, and A. Gacic. Multisensor Fusion in Smartphones for Lifestyle Monitoring. In *Body Sensor Networks (BSN), 2010 International Conference on*, pages 36–43. IEEE, 2010.
[10] J. Garcia, J. Molina, A. Berlanga, and G. de Miguel. Data fusion alternatives for the integration of millimetre radar in airport surveillance systems. In *Radar Conference, 2005 IEEE International*, pages 796–801. IEEE, 2005.
[11] M. Henriksen, H. Lund, R. Moe-Nilssen, H. Bliddal, and B. Danneskiod-Samsoe. Test-retest reliability of trunk accelerometric gait analysis. *Gait & posture*, 19(3):288–297, 2004.
[12] P. Korpipaa, J. Mantyjarvi, J. Kela, H. Keranen, and E. Malm. Managing context information in mobile devices. *Pervasive Computing, IEEE*, 2(3):42–51, 2003.
[13] N. Lane, E. Miluzzo, H. Lu, D. Peebles, T. Choudhury, and A. Campbell. A survey of mobile phone sensing. *Communications Magazine, IEEE*, 48(9):140–150, 2010.
[14] L. Liao. *Location-based activity recognition*. PhD thesis, Citeseer, 2006.
[15] M. Marin-Perianu, N. Meratnia, P. Havinga, L. de Souza, J. Muller, P. Spiess, S. Haller, T. Riedel, C. Decker, and G. Stromberg. Decentralized enterprise systems: a multiplatform wireless sensor network approach. *Wireless Communications, IEEE*, 14(6):57–66, 2007.
[16] M. Markin, C. Harris, M. Bernhardt, J. Austin, M. Bedworth, P. Greenway, R. Johnston, A. Little, and D. Lowe. Technology foresight on data fusion and data processing. *Publication of The Royal Aeronautical Society*, 1997.
[17] E. Miluzzo, N. Lane, K. Fodor, R. Peterson, H. Lu, M. Musolesi, S. Eisenman, X. Zheng, and A. Campbell. Sensing meets mobile social networks: the design, implementation and evaluation of the cenceme application. In *Proceedings of the 6th ACM conference on Embedded network sensor systems*, pages 337–350. ACM, 2008.
[18] M. Mun, S. Reddy, K. Shilton, N. Yau, J. Burke, D. Estrin, M. Hansen, E. Howard, R. West, and P. Boda. Peir, the personal environmental impact report, as a platform for participatory sensing systems research. In *Proceedings of the 7th international conference on Mobile systems, applications, and services*, pages 55–68. ACM, 2009.
[19] D. Recordon and D. Reed. OpenID 2.0: a platform for user-centric identity management. In *Proceedings of the second ACM workshop on Digital identity management*, pages 11–16. ACM, 2006.
[20] B. Schilit and M. Theimer. Disseminating active map information to mobile hosts. *Network, IEEE*, 8(5):22–32, 2002.
[21] A. Schmidt, K. Aidoo, A. Takaluoma, U. Tuomela, K. Van Laerhoven, and W. Van de Velde. Advanced interaction in context. In *HandHeld and Ubiquitous Computing*, pages 89–101. Springer, 1999.
[22] A. Schmidt, M. Beigl, and H. Gellersen. There is more to context than location. *Computers & Graphics*, 23(6):893–901, 1999.
[23] R. Suoranta and A. Lappetel
"ainen. Operatoris Dilemma. 2010.
[24] A. Thiagarajan, L. Ravindranath, K. LaCurts, S. Madden, H. Balakrishnan, S. Toledo, and J. Eriksson. VTrack: accurate, energy-aware road traffic delay estimation using mobile phones. In *Proceedings of the 7th ACM Conference on Embedded Networked Sensor Systems*, pages 85–98. ACM, 2009.
[25] F. van Diggelen. *A-GPS: Assisted GPS, GNSS, and SBAS*. Artech House Publishers, 2009.
[26] R. Want. You are your cell phone. *Pervasive Computing, IEEE*, 7(2):2–4, 2008.